QUARTERLY R&D STATUS REPORT - January, 1991

Reporting Period 10

ARPA Order Number:	6419
Program Code Number:	8E20
Contractor:	Cornell University
Contract Number	N00014-88-K-0591
Contract Amount:	$2,931,739
Effective Date of Contract:	July 1, 1988
Expiration Date of Contract:	June 30, 1991

Using Computer Design and Simulation to Improve Manufacturing Productivity

Principal Investigator:	John E. Hopcroft
Telephone Number:	607-255-7316
Electronic Mail Address:	jeh@gvax.cs.cornell.edu

DESCRIPTION OF PROGRESS

1. Environments for Scientific Computation

Our work on Simulaid, our simulator generation environment, is continuing. During the past quarter, we have implemented a prototype simulation environment and used it to construct a simple simulation. The simulator is specified in a few pages of text, defining the mathematical model and the method for solving it. The Simulaid system, given this specification as input, constructs a simulator for that problem.

The first simulator we specified solves a simple fluid flow problem. The mathematical model contains definitions for the primitive objects in this simulator (fluids, boundary sections, and boundary conditions), as well as the equation used to model the flow (in this case Laplace's equation). The simulation specification, which provides a method for interpreting the mathematical model, uses the boundary element method to solve the partial differential equation defined in the model.

2. Finite Element Mesh Generation

In recent work, we have applied our mesh generation methods to the problem of adaptive remeshing in 2D, with a goal of using these techniques in the analysis of a portion of a diesel engine. We have enclosed a simple example showing adaptive remeshing for a problem involving steady-state heat distribution on a flat plate containing an oddly-shaped hole.

3. Robust Geometric Algorithms

We continue to investigate the design of robust geometric algorithms like those used in solid modelers and finite-element mesh generators. Current algorithms assume perfect, infinite precision, arithmetic and ignore the problems of finite precision arithmetic. These algorithms fail because of roundoff error and inaccurate input.

In last quarter's report the notion of an *approximate polygon* was introduced to model polygons whose positions were uncertain. One important question asked in solid modeling is whether a point lies in a polygon. Using the idea of an approximate polygon, we have developed a robust algorithm which answers this question in a consistent manner. The robust point location algorithm is fast and simple, and can handle uncertainty in both the point location and the polygon location.

We are also interested in *local correctness*. A *locally correct* algorithm always succeeds when presented with localized special cases. For example, a locally correct intersection algorithm always succeeds when intersecting two corners, two edges, or two faces. In the last report we concluded that global

correctness is nearly impossible to achieve in a robust polyhedral intersection algorithm, and that proving local correctness would be the best alternative.

The last report gave several rules which, when incorporated in an intersection algorithm, ensure local correctness for corners. Since then, rules have been developed to ensure local correctness for intersecting two edges. As soon as face rules have been developed, a robust algorithm for polyhedron intersection will be implemented.

4. Programming the Control of Complex Systems

We have worked on using automatic differentiation to speed up the control of high degree of freedom mechanical systems such as our biped walker. Automatic differentiation is a technique for computing all the derivatives of a factorable function, which requires only a small factor more (≤ 5 times for rational functions) the time to compute the function itself. Our technique of programming complex mechanical systems uses a version of Newton's method for satisfying the numerous constraints which arise; therefore it is important to be able to compute the gradients of the constraint functions efficiently. We have implemented a program to parse lisp expressions into a Kantorovich graph and to augment the graph so that the gradients can be computed directly. Users are freed from the burden of coding much of the kinematic transformations and their derivatives, while still producing efficiently executable control programs. A secondary advantage of the graph representation is that it is a good intermediate representation which can be compiled to a straight line program in a language like C or assembler, enabling us to directly transfer working programs from simulations to actual mechanical hardware. We can currently compile the graph to straight line lisp code. In the near future, we hope to use the representation to analyze data dependencies in control computations, leading to parallel programs. We are also exploring the use of automatic differentiation to differentiate the Lagrangian of a constrained mechanical system, possibly leading to a more efficient formulation of the dynamics.

List of Publications

1. "Free-form Surface Modeling Using Implicit Patches." Proceedings of the Second Canadian Conference on Computational Geometry, Ottawa, Ontario, August, 1990. Baining Guo.

2. "The Global Solution to a Class of Multidimensional Nonlinear Wave Equations." Chinese Annals of Mathematics, Vol. 5, Ser.A No,4 1984. Baining Guo.

3. "Automatic Surface Generation Using Implicit Cubics." Proceedings of the Ninth International Conference on Computer Graphics, Computer Graphics International '91, Cambridge, MA. June, 1991. Baining Guo.

4. "Masking Failures of Multidimensional Sensors." Submitted to the Seventh Annual Symposium on Computational Geometry. L. Paul Chew and Keith Marzullo.

5. "A Plane-Sweep Algorithm for Exact Simulation of a Quasi-Static Planar Mechanical System of Compliantly Connected Rigid Bodies." Submitted to 1991 ACM Symposium on Computational Geometry. Dinesh Pai (with B. R. Donald.)

Presentations

1. "An Efficiently Computable Metric for Comparing Polygonal Shapes." First Annual Symposium on Discrete Algorithms, San Francisco, January 1990. L. Paul Chew.